The ternary operator

In languages like C and Java, there is a pattern that often occurs in programs:

```
// Example 1
if (age >= 18)
   price = 30;
else
   price = 25;

// Example 2
if (turningRight)
   angle = 90;
else
   angle = -90;

// Example 3
if (value.equals("1"))
   b = true;
else
   b = false;
```

The common pattern in these examples is “set a variable to some value if a condition holds, otherwise set it to another value.” We can abstract it as follows:

```
if ( <condition> )

     <name> = <value1> ;

else

     <name> = <value2> ;
```

This pattern is so often used that the designers of the C language have created a shorthand construct for this, that is also available in C++ and Java.

The conditional expression

Syntax:

> <condition> ? <expression1> : <expression2>
>
> (a conditional expression, followed by a question mark, followed by an expression, followed by a colon “:”, followed by another expression)

Semantics:

> The expression on the left is evaluated first. If it evaluates to "true", the expression in the middle is evaluated and its result is returned. Otherwise, the expression on the right is evaluated and its result is returned.

For example:

```
price = (age >= 18) ? 30 : 25;

angle = turningRight ? 90 : -90;

b = value.equals("1") ? true : false;
```

This construct abstracts over the general pattern of conditional assignment to a variable. It is even slightly more powerful, because we can nest it!

For example:

```
price = (age <= 6) ? 5 : ((age <= 18) ? 25 : 30);

// is equivalent to:
if (age <= 6)
   price = 5;
else
   if (age <= 18)
      price = 25;
   else
      price = 30;
```

This construct is called the *conditional expression*, not to be confused with the conditional statement that uses "`if`".

We name:

- "conditional expression" an instance of the construct, when the three expressions are filled in;
- *"conditional operator"* the punctuation characters "`?:`".

In other words the conditional expression combines three expressions with the conditional operator.

Ternary operator

The conditional operator is also called *ternary operator* because it has three positions. It is called "the" ternary operator in C and Java because it is the only operator with 3 positions in the language.

Important concepts

- definition of the *conditional expression*;
- difference between conditional expression (using "?:") and conditional statement (using "`if`");
- difference between conditional expression and conditional operator
- the alternative name *"ternary operator"* for the conditional operator

Further reading

- Introduction to Programming, section 2.5.5 (p. 51)
- Absolute Java, section 3.1 (pp. 112-113)

Copyright and licensing